AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application.

Listing of Claims:

1. (Currently Amended) A method of addressing data errors in a computer system, comprising:

pre-determining a software-programmable data poisoning policy to control actions to be taken based on different classes of data poisoning events by a user;

error-checking a unit of data by an error-control decoder;

detecting an uncorrectable error in the unit of data by the error-control decoder;

if the uncorrectable error is detected in the unit of data, based on the pre-determined data poisoning policy, determining if the detected uncorrectable error is a data poisoning event;

marking the unit of data containing a data poisoning event with a software-visible bit by the error-control decoder which software-visible bit is a status bit to indicate to an operating system that the data unit contains the data poisoning event;

determining, based on the pre-determined data poisoning policy, if the unit of data containing the poisoning event is to be acted upon;

handing over the data units including the detected uncorrectable errors including data poisoning events from the error-control decoder to the operating system;

detecting by [[an]] the operating system whether the software-visible bit is present in the data unit; and

~~in accordance with~~ based on the detected software-visible ~~control~~ bit and the pre-determined data poisoning policy, acting, by the operating system of the computer system, to address the presence of the uncorrectable error in the unit of data, including:

if the software-visible bit is detected, one of removing the marked data unit from a use by the operating system and recovering the data unit or determining if the unit of data is in user space and one of terminating an application running on the computer system and removing the unit of data from use by the operating system or shutting down the operating system, and

if the software-visible bit is not detected, determining whether the data unit is in the user space and one of terminating the application running on the computer system and removing data unit from use by the operating system or shitting down the operating system ~~is not always brought down~~.

2. (Previously Presented) The method of Claim 1, wherein error-checking comprises:
applying error-control decoding to the unit of data.

3. (Previously Presented) The method of Claim 2, wherein error-checking further comprises:
correcting correctable errors in the unit of data.

4-8. (Cancelled)

9. (Original) The method of Claim 1, further comprising:
upon detection of an uncorrectable error in said unit of data, providing information to said operating system to enable recovery of said unit of data.

10. (Original) The method of Claim 9, wherein the information includes a target address corresponding to said unit of data.

11-12. (Cancelled)

13. (Previously Presented) The method of Claim 1, wherein detecting is performed by at least one of a processor or a memory.

14. (Currently Amended) A computer system comprising:
a software-programmable data poisoning policy to control actions to be taken based on different classes of data poisoning events;
a processor;

at least one error control decoding implementation including ~~at least one of~~ an error-control decoder~~, a software to implement error-control decoding by the processor, or a firmware~~ to implement error-control decoding in conjunction with the processor, adapted to process units of data and to detect if a unit of data contains at least one uncorrectable error; and

a module to run on said processor to determine, based on the pre-determined data poisoning policy, if said uncorrectable error is a data poisoning event, and, if so, to mark unit of data containing said uncorrectable error and hand over the data units including the detected uncorrectable errors including data poisoning events from the error-control decoder to [[and]] an operating system which runs ~~to run~~ on said processor, the operating system to determine, based on the pre-determined data poisoning policy, if a particular data poisoning event is to be acted upon or not, the operating system adapted to detect the marked unit of data and to act to mitigate the detected uncorrectable error without always bringing down the operating system upon detection of the marked unit of data.

15. (Previously Presented) The computer system of Claim 14, further comprising:

a memory coupled to said error control decoding implementation, wherein the error-control decoding implementation is adapted to process units of data stored in the memory.

16. (Previously Presented) The computer system of Claim 15, wherein said memory comprises:

a processor cache.

17. (Previously Presented) The computer system of Claim 14, further comprising:

at least one bus coupled to said error-control decoding implementation, wherein the error-control decoding implementation is adapted to process units of data passing through the bus.

18. (Original) The computer system of Claim 14, further comprising:

logic adapted to control signaling of information relating to one or more uncorrectable data errors.

19. (Original) The computer system of Claim 18, wherein the logic comprises:

programmable logic.

20. (Previously Presented) The computer system of Claim 18, wherein the information includes a target address corresponding to a unit of data containing the detected uncorrectable error.

21. (Currently Amended) A machine-accessible storage medium containing software code that, when read by a computer, causes the computer to perform a method comprising:

by a user, pre-determining a software-programmable data poisoning policy to control actions to be taken based on different classes of data poisoning events;

error-checking a unit of data by an error-control decoder;

if an uncorrectable error is detected in the unit of data, based on the pre-determined data poisoning policy, determining if the detected uncorrectable error is a data poisoning event, and if so, marking the unit of data containing a data poisoning event with a software-visible bit by the error-control decoder;

determining, based on the pre-determined data poisoning policy, if the unit of data containing poisoning event is to be acted upon, and if so, handing over the data units including the detected uncorrectable errors including data poisoning events from the error-control decoder to an operating system;

detecting, by [[an]] the operating system of the computer, the software-visible bit in the unit of data; and

in accordance with the detected software-visible control bit and the pre-determined data poisoning policy, acting, by the operating system of the computer to address the presence of the uncorrectable error in the unit of data, wherein the operating system is not always brought down.

22. (Previously Presented) The machine-accessible storage medium of Claim 21, further comprising software code that, when read by a computer, causes the computer to also perform the following:

if the operating system detects the software-visible bit, determining if the unit of data is in user space; and

if the unit of data is in user space, terminating an application running on the computer and removing the unit of data from use by the operating system.

23. (Previously Presented) The machine-accessible storage medium of Claim 21, wherein said acting upon the presence of the uncorrectable error comprises:

removing the unit of data from use by the operating system.

24. (Currently Amended) A computer system comprising:

a processor; and

machine-accessible storage medium to be coupled to the processor, the processor to access the machine-accessible storage medium and to execute software code stored on the machine-accessible storage medium, to cause the computer system to perform a method comprising:

by a user, pre-determining a software-programmable data poisoning policy to control actions to be taken based on different classes of data poisoning events;

error-checking a unit of data by an error-control decoder;

if an uncorrectable error is detected in the unit of data, based on the pre-determined data poisoning policy, determining if the detected uncorrectable error is a data poisoning event, and if so, marking the unit of data containing a data poisoning event with a software-visible bit by the error-control decoder;

determining, based on the pre-determined data poisoning policy, if the unit of data containing poisoning event is to be acted upon, and if so, handing over the data units including the detected uncorrectable errors including data poisoning events from the error-control decoder to an operating system and detecting, by [[an]] the operating system of the computer system, the software-visible bit in the unit of data; and

in accordance with the detected software-visible control bit and the pre-determined data poisoning policy, acting, by the operating system to address the presence of the uncorrectable error in the unit of data, wherein the operating system is not always brought down.

25. (Previously Presented) The computer system of Claim 24, wherein the machine-accessible storage medium further comprises software code that, when executed by the processor, causes the computer system to further perform:

if the operating system detects the software-visible bit, determining if the unit of data is in user space; and

if the unit of data is in user space, terminating an application running on the computer system and removing the unit of data from use by the operating system.

26. (Previously Presented) The computer system of Claim 24, wherein the machine-accessible storage medium further comprises software code that, when executed by the processor, causes the computer system to further perform:

removing the unit of data from use by the operating system.

27. (Previously Presented) The computer system of Claim 24, further comprising:

at least one bus to couple the processor with the machine-accessible storage medium.